Claims 1-31 and 33-44 are pending bust stand rejected. Claims 1, 8-14, 24, 25, 31, 33, 34, 35, and 40 have been amended. Claim 32 was previously cancelled. In view of the following remarks, the Applicant respectfully requests that the Examiner thoughtfully reconsider and pass the application on to issuance.

CLAIM REJECTIONS – 35 USC §103: The Examiner rejected Claims 1-31 and 33-44 as being as unpatentable over USPN 6,163,806 issued to Viswanathan in view of USPN 6,931,105 issued to Groves.

Claim 1 is directed to a method for creating a best-match object at run time and, as amended, recites the following:

1. receiving a request for an object;

2. polling a plurality of object proxies for a confidence level representing the capability of each respective proxy to generate the requested object, each object proxy representing and configured to create a particular object;

3. receiving, from each of the plurality of object proxies, a confidence level indicative of that object proxy's ability to generate the requested object;

4. selecting one of the proxies based on the received confidence levels; and

5. directing the selected proxy to create the object represented by the selected object proxy.

Viswanathan discloses a system that provides access to "devices on a computer cluster." Viswanathan, abstract. When a user requests that a particular device on the

cluster be opened, an operating system kernel identifies information such as the device's node and local address and then issues an "open request to the host node for the device." Viswanathan, abstract. A special file system operating on the host node returns a "special file object that is associated with the desired device." Viswanathan, abstract.

The following passage from Viswanathan explains further:

As described above, the kernel 242 handles system calls from the applications 150, such as requests to access the memory 230, the file system 206 or the devices 106. The kernel 242 differs from the kernel 132 (FIG. 1) as it has been modified by the present invention to support global device access. The proxy file system (PxFS) 244 is based on the Solaris PxFS file system but, like the kernel 242, is modified herein to support global device access. *The PxFS 244 includes a collection of objects that enable an application 150-i in one node 202-i to interact seamlessly with the file system 206 across different nodes 202. The PxFS objects include PxFS clients 246, PxFS servers 248, f__objs (file objects) 250, vnodes (virtual file nodes) 252, snodes (special file nodes) 254 and px__vnodes (proxy vnodes) 256. Each of these objects is labeled in FIG. 6 as optional (opt) as they are created as needed by the PxFS 244 in response to operations of the file system 206.*

Viswanathan, col. 9, lines 43-59. As a further example, Viswanathan explains:

An object such as the kernel 242 that needs to access the file system 206 first issues the access request to its local PxFS client 246. The PxFS client holds a reference to the PxFS server 248 co-located with the file system 206. This reference enables the PxFS client 246 to communicate the kernel's request to the file system 206 via the PxFS server 248. The file system 206 performs the requested access, creates a vnode object 252 representing the requested file and returns a reference to vnode object 252 to the PxFS server 248. Because the nodes 202-1 and 202-2 are different address spaces, the reference to the vnode 252 is useless to the PxFS client 246 and kernel 242 in the node 202-1. Consequently, the PxFS server 248 creates a file transport object (f__obj) 250 linked to the vnode 252 and returns a reference to the f__obj 150 to the PxFS client 246. Upon receiving the f__obj reference the PxFS client 246 creates a proxy vnode (px__vnode) 256 that is linked to the f__obj

250. The kernel 242 can then access the file information represented by
the vnode 252 by simply accessing the local px__vnode 256.

Viswanathan, col. 16, lines 32-51.

Viswanathan teaches an operating system kernel that communicates with a
proxy file system (PxFS) on behalf of an application. The proxy file system includes a
collection of objects that allow the application to communicated with a remote file
system. While not clear, it appears that the Examiner is equating the kernel's
communication with the proxy file system with polling object proxies. Claim 1 has been
amended to clarify that each of a plurality of object proxies are polled. Viswanathan, at
best, teaches communication with a single proxy file system and mentions nothing of
polling or otherwise communicating with a plurality of proxy file systems.

Claim 1 has also been amended to recite receiving, from each of the plurality of
object proxies, a confidence level indicative of that object proxy's ability to generate the
requested object. As noted, Viswanathan mentions nothing of an interaction with
plurality of object proxies. As such Viswanathan mentions nothing of receiving
information of any type from a plurality of object proxies.

The Examiner admits that Viswanathan fails to teach or suggest the use or
determination of confidence levels and thus fails to teach or suggest (a) receiving, from
each of the plurality of object proxies, a confidence level indicative of that object proxy's
ability to generate the requested object and (b) selecting one of the proxies based on
the received confidence levels.

The Examiner asserts that "Groves teaches selecting a proxy-server address
data based on polled confidence level and internal goodness of a best match that is
determined/created if absent with the caller spoken input as disclosed in the abstract of
his invention (abstract; col. 3, lines 14-30; col. 4, lines 1-5 and 36-46)." Groves is
describes a system that correlates call data and speech recognition information.
Groves, title and abstract. Groves defines call data to include Call data may include
caller-ID information, automatic number identification (ANI) information, dialed number

information system (DNIS) information, an Internet address such as an IP address associated with the caller 100, a truncated Internet address (for example, only a domain), an alias, and a proxy-server address. Groves, col. 3, lines 31-40 and col. 3, line 58 through col. 4, line 5. Groves teaches that when confidence in voice recognition of spoken input falls below a threshold, the call data can be used in correlation to identify alphanumeric information that best represents the spoken input. Groves, abstract. The Examiner's assertion that Groves teaches selecting a proxy server address based on a confidence level is simply not true. Groves teaches using a proxy server address to identify alphanumeric information.

Even if the Examiner's assertions were true, Groves even when combined with Viswanathan does not teach (a) receiving, from each of the plurality of object proxies, a confidence level indicative of that object proxy's ability to generate the requested object and (b) selecting one of the proxies based on the received confidence levels. For at least these reasons, Claim 1 and Claims 2-7 which depend from Claim 1 are patentable over Viswanathan and Groves.

Claim 8 recites a system that includes an object generator and a processor operable to execute the object generator. The object generator includes instructions that when executed by the processor function as:

1.   means for receiving indicia of an object to be created;

2.   means for polling a plurality of object proxies for a confidence level representing the capability of each respective proxy to generate the requested object, each object proxy representing and configured to create a particular object;

3.   means for receiving, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object;

4.   means for selecting one of the proxies based on the ~~polled~~ received confidence levels; and

5.   means for directing the selected proxy to create the object represented by the selected object proxy.

As discussed above Viswanathan and Groves fail to teach or suggest (a) receiving, from each of the plurality of object proxies, a confidence level indicative of that object proxy's ability to generate the requested object and (b) selecting one of the proxies based on the received confidence levels. For the same reasons, those references fail to teach or suggest (a) means for polling a plurality of object proxies for a confidence level representing the capability of each respective proxy to generate the requested object, each object proxy representing and configured to create a particular object and (b) means for receiving, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object.

For at least these reasons, Claim 8 and Claims 9-13 which depend from Claim 8 are patentable over Viswanathan and Groves.

Claim 14 recites a system that includes an object generator and a processor operable to execute the object generator. The object generator includes instructions that when executed by the processor function as:

1.   a pool including a plurality of object proxies, each object proxy configured to create a respective object, the pool configured to receive indicia of a requested object and each of the plurality of object proxies configured to return a respective confidence level indicative of that object proxy's ability to generate the requested object; and

2.   an object factory configured to:

      a.   poll the plurality of object proxies for a confidence level representing the capability of each respective proxy to generate the requested object,

b. to receive, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object,

c. to select one of the proxies based on the received confidence levels, and

d. to direct the selected object proxy to create the object represented by the selected object proxy.

As discussed above Viswanathan and Groves fail to teach or suggest polling a the plurality of object proxies for a confidence level representing the capability of each respective proxy to generate the requested object and receiving to receive, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object. For at least these reasons, Claim 14 and Claims 12-23 which depend from Claim 14 are patentable over Viswanathan and Groves.

Claim 24 is directed to a computer readable medium that includes logic configured to implement the method of Claim 1. For at least the same reasons Claim 1 is patentable, so are Claim 24 and Claims 25-30 which depend from Claim 24.

Claim 31 recites a system that includes an object generator and a processor operable to execute the object generator. The object generator includes instructions that when executed by the processor function as:

1. an object factory configured to receive a device identifier;

2. a pool having an interface configured to communicate with the object factory, the pool containing a plurality of object proxies each representing and capable of producing a respective object; and

3.   an object store coupled to the pool and configured to receive and retain objects generated by selected object proxies;

4.   wherein the object factory is configured to poll each of the plurality of object proxies for a confidence level representing the capability of the respective object proxy to generate an object suited for operating with a device responsive to the device identifier, to receive, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object, to select one of the plurality of proxies responsive to the received confidence levels, and to direct the selected proxy to create the object represented by the selected object proxy.

As discussed above Viswanathan and Groves fail to teach or suggest polling a plurality of object proxies for a confidence level representing the capability of each respective proxy to generate the requested object and receiving to receive, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object. For the same reasons, those references fail to teach or suggest an object factory that is configured to poll each of the plurality of object proxies for a confidence level representing the capability of the respective object proxy to generate an object suited for operating with a device responsive to the device identifier and to receive, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object. For at least these reasons, Claim 31 is patentable over Viswanathan and Groves.

Claim 33 is directed to a method for creating a best-match object at run time and recites the following:

1.   loading a plurality of object proxies;

2.   receiving indicia of a desired object for communicating with a peripheral device;

3.   directing each of the plurality of object proxies to forward a confidence level representing the capability of each respective object proxy to generate the desired object;

4.   for each of the plurality of object proxies, receiving a confidence level associated with that object proxy

5.   comparing the received confidence levels to a maximum confidence level;

6.   if a first one of the received confidence levels matches the maximum confidence level, directing the object proxy associated with the first one of the confidence levels to generate an object; and

7.   if none of the received confidence levels matches the maximum confidence levels identifying a second one of the received confidence levels that exceeds the remaining ones of the received confidence levels and directing the object proxy associated with the second one of the confidence levels to generate an object.

As discussed above Viswanathan and Groves fail to teach or suggest polling a plurality of object proxies for a confidence level representing the capability of each respective proxy to generate the requested object and receiving to receive, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object. For the same reasons, those references fail to teach or suggest a method that includes (a) directing each of the plurality of object proxies to forward a confidence level representing the capability of each respective object proxy to generate the desired object and (b) for each of the plurality of object proxies, receiving a confidence level associated with that object proxy. For at least these reasons, Claim 33 is patentable over Viswanathan and Groves.

Claim 34 is directed to a computer readable medium that includes logic configured to implement the method of Claim 33. For at least the same reasons Claim 33 is patentable, so is Claim 34.

Claim 35 is directed to a method for creating a best-match printer driver and recites the following:

1.   receiving a request to use a printer;

2.   polling a plurality of printer driver proxies for a confidence level representing the capability of each respective printer driver proxy to generate a driver that when applied to data and forwarded to the printer will produce a useful representation of the data, each printer driver proxy representing and configured to create a particular printer driver;

3.   receiving, from each of the plurality of printer driver proxies, a confidence level indicative of that printer driver proxy's ability to generate a driver for the printer;

4.   selecting one of the printer driver proxies based on the received confidence levels; and

5.   directing the selected printer driver proxy to generate the driver represented by the selected printer driver proxy.

As discussed above Viswanathan and Groves fail to teach or polling a plurality of object proxies for a confidence level representing the capability of each respective proxy to generate the requested object and receiving to receive, from each of the object proxies, a confidence level indicative of that object proxy's ability to generate the requested object. For the same reasons, those references fail to teach or suggest a method that includes the act of polling and receiving as recited in Claim 35. For at least these

reasons, Claim 35 and Claims 36-39 which depend from Claim 34 are patentable over Viswanathan and Groves.

Claim 40 is directed to a computer readable medium that includes logic configured to implement the method of Claim 35. For at least the same reasons Claim 35 is patentable, so is Claim 40 and Claims 41-44 which depend from Claim 40..

CONCLUSION: The foregoing is believed to be a complete response to the outstanding Office Action. Claims 1-31 and 33-44 are all felt to be in condition for allowance. Consequently, early and favorable action allowing these claims and passing the application to issue is earnestly solicited. The foregoing is believed to be a complete response to the outstanding Office Action.

Respectfully submitted,
John L. Manuel

By /Jack H. McKinney/
Jack H. McKinney
November 12, 2007   Reg. No. 45,685